

---

# Network Mode standard

## Error probability and timeouts study

---

Group A, B and C



# Table of contents

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Introduction</b>   | <b>2</b>  |
| 1.1      | Scope . . . . .   | 2         |
| <b>2</b> | <b>Error probability</b>                                    | <b>3</b>  |
| 2.1      | Required number of retries . . . . .                        | 3         |
| 2.2      | Bit error probability $p_{eb}$ . . . . .                    | 4         |
| 2.3      | Numerical example for the NM competition scenario . . . . . | 5         |
| 2.3.1    | DATA packet, neighbour distance . . . . .                   | 5         |
| 2.3.2    | DATA packet, 3 nodes distance . . . . .                     | 7         |
| 2.3.3    | TOKEN packet, neighbour distance . . . . .                  | 8         |
| 2.3.4    | TOKEN packet, 3 nodes distance . . . . .                    | 9         |
| 2.3.5    | Summary . . . . .   | 9         |
| <b>3</b> | <b>Timeouts</b>   | <b>11</b> |
| 3.1      | Theoretical explanation . . . . .                           | 11        |
| 3.2      | Numerical example for the NM competition scenario . . . . . | 12        |

# 1. Introduction

This document has been written with the purpose to quantify the error probability that transceivers that implement the Network Mode standard can suffer, as well as to study the minimum timeout requirements. A proper analysis can lead to optimal values than enhance the performance without hurting the robustness of the standard.

## 1.1. Scope

This document establishes the minimum number of retries and the minimum timeout to achieve a successful Network Mode standard operation. It starts from generic explanations and equations to later focus on the Network Mode particular case. For both chapters there are, first, theoretical sections; and after a section that shows calculations for a concrete application.

This document is applicable to transceivers that implement the Network Mode standard in its totality. Numerical results given as examples are made for the Network Mode competition constraints; the reader should perform his own calculations if he brings the Network Mode standard to other scenarios.

## 2. Error probability

### 2.1. Required number of retries

The bit error rate (BER) can be understood as the expectancy for a bit, sent by a transmitter, to be flipped at the receiver side, by unit of time. To avoid time dependency this document makes use of bit error probability, defined as  $p_{eb}$ . If multiple bits are exchanged between a transmitter and a receiver, then the following holds:

$$p_{eb} = E \left\{ \frac{\# \text{ erroneous bits}}{\# \text{ total bits}} \right\} = \frac{E \{ \# \text{ erroneous bits} \}}{\# \text{ total bits}} . \quad (1)$$

The Network Mode standard defines how packets of bits are exchanged between nodes. Here, a packet is considered erroneous if at least one of its bits is incorrect. Then, the probability for a packet to have at least one error can be understood as a particular binomial distribution case in which the number of trials is equal to the number of trials in which the bit hasn't been flipped.

$$P(x)|_{x=n} = \binom{n}{x} p^x (1-p)^{n-x} = p^x . \quad (2)$$

Where  $p$  is the event probability,  $n$  the number of trials and  $x$  the number of trials for which the event occurs.

The probability for a packet to differ, at least by 1 bit, with respect to the one at the transmitter side is defined as  $p_{ep}$ . For  $N$  bits in a packet, with the bit error probability being  $p_{eb}$ ,  $p_{ep}$  is defined as 1 minus the probability of having  $N$  out of  $N$  bits correct.

$$p_{ep} = 1 - (1 - p_{eb})^N . \quad (3)$$

It's already conceived in the standard that some received packets may contain errors, and thus the packet can be resent, i.e., a retry is made from the transmitter side. The probability to fail after  $T$  tries is named  $p_{\text{fail}}$  and follows

$$\begin{aligned} p_{\text{fail}} &= (p_{ep})^T \\ p_{\text{fail}} &= \left( 1 - (1 - p_{eb})^N \right)^T . \end{aligned} \quad (4)$$

It's a goal of the standard to achieve a robust communication, which means the probability of failing at sending a certain packet of  $N$  bits after  $T$  tries should be lower than a desired threshold.

$$p_{\text{fail}} = \left( 1 - (1 - p_{eb})^N \right)^T < p_{\text{fail threshold}} . \quad (5)$$

The number of packet bits  $N$  is fixed by the defined packets, and  $p_{eb}$  can be derived after link budget calculations. The fail probability  $p_{\text{fail threshold}}$  is set to an arbitrary value. The unknown is the number of tries required,  $T$ , which can only take integer values equal or higher than 1. Because the number inside the parenthesis must always be lower than 1,  $T$  must respect a

minimum value.

$$T > \frac{\ln(p_{\text{fail threshold}})}{\ln(1 - (1 - p_{eb})^N)} . \quad (6)$$

The number of retries is just

$$R = T - 1 . \quad (7)$$

## 2.2. Bit error probability $p_{eb}$

RF modules manufacturers typically provide the expected bit error probability (usually named as BER) for a single received power. To solve (6) for  $T$ , link budget calculations are required, and the calculated received power signal at the receiver end should be used to determine  $p_{eb}$ .

Complex equations could be used to express  $p_{eb}(E_b/N_0)$ , where  $E_b$  is the bit energy and  $N_0$  is two times the noise added by the channel; but this job has already been done by Luque, Morón, and Casilari [1]. In the paper, the authors study a Bluetooth system, which works with Gaussian frequency-shift keying (GFSK). The Network Mode proposed standard uses GFSK. The bit error probability depends on the modulation index  $h$ . The  $p_{eb}$  is plotted against the  $E_b/N_0$  (dB) ratio in Figure 1.

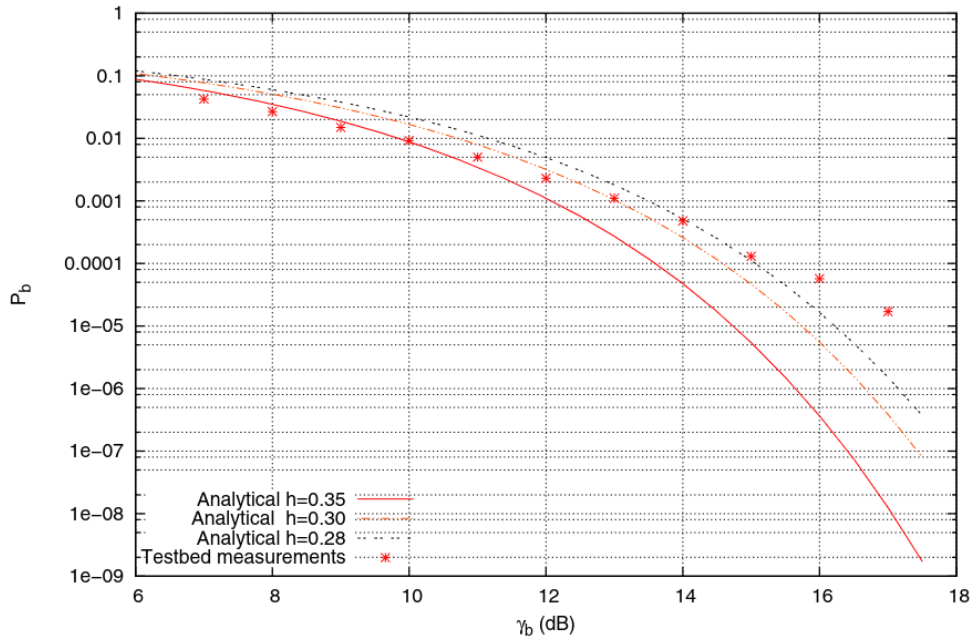


Figure 1.  $p_{eb}$  as a function of  $E_b/N_0$  (dB), from Luque, Morón, and Casilari [1]

The standard doesn't specify a concrete  $h$  value. The testbed measurements will be used for future calculations because it's the worst case for large  $E_b/N_0$  values and shows quite a linear trend in the logarithmic axes.

The number of symbols of a codification, named  $M$ , follows

$$M = 2^{R_b/BW} , \quad (8)$$

where  $BW$  is the bandwidth and  $R_b$  the bandwidth efficiency. In GFSK there are only two symbols possible: 0 or 1. This means  $R_b = BW$ .

The signal-to-noise ratio  $SNR$  at the output of the receiver demodulator is defined in the paper as

$$SNR = \frac{E_b}{N_0 \cdot BW \cdot T_b}, \quad (9)$$

where  $T_b = \frac{1}{R_b}$  is the bit transmission period. Substituting (8) leads to

$$SNR = \frac{E_b}{N_0}, \quad (10)$$

for the particular GFSK case. By definition the receiver sensitivity is the power level for which a certain received signal leads to a certain BER, and this means

$$SNR = \frac{E_b}{N_0} [\text{dB}] = P_{RX} [\text{dBm}] - (P_{RX, \text{sensitivity}} [\text{dBm}] - P_{\text{offset}} [\text{dBm}]), \quad (11)$$

where  $P_{\text{offset}}$  is the power difference between the noise floor and the sensitivity that has been specified for a certain BER smaller than 0.5.

In order to find  $P_{RX}$  link budget calculations are necessary. Assuming quite and ideal case with line of sight (LOS) and no ground reflections,

$$P_{RX} [\text{dBm}] = P_{TX} [\text{dBm}] + G_{\text{antenna TX}} [\text{dB}] - 20 \log \left( \frac{4\pi d}{\lambda} \right) [\text{dB}] + G_{\text{antenna RX}} [\text{dB}] - G_{\text{losses}} [\text{dB}]. \quad (12)$$

Where  $d$  is the distance between antennas, and the gain that antennas can offer and extra possible losses has been considered.

Using the previous definitions (6) can be solved. As a summary,  $P_{RX}$  should be calculated from the link budget, and  $P_{RX, \text{sensitivity}}$  can be either calculated from LNA noise figure (NF) and input thermal noise, or taken from the receiver manufacturer specifications. Then, if  $P_{\text{offset}}$  is known it's straightforward to get  $\frac{E_b}{N_0}$ . Next, a plot such as the one in Figure 1 can be used to get the bit error probability,  $p_{eb}$ . Finally, this value is substituted in (6) together with the  $N$  bits of the packet and an arbitrary  $p_{\text{fail}}$  threshold,

### 2.3. Numerical example for the NM competition scenario

The purpose of this section is to provide some application-dependant orientative numbers to solve for  $T$ .

#### 2.3.1. DATA packet, neighbour distance

First, for the link budget it's taken into account that nRF24 output power is set to  $-6 \text{ dBm}$ . The nRF24l01 contains a low noise amplifier (LNA) that provides  $10 \text{ dB}$ . The sensitivities

given by the manufacturer are referenced to the LNA output, meaning that at the LNA input they are 10 dB lower. Both the transmitter and the receiver antennas show the same gain:  $G_{\text{antenna RX}} = G_{\text{antenna TX}} = 2 \text{ dB}$ .  $G_{\text{losses}}$  is set to 6 dB to account for the ground reflection effect.

It's assumed 4 transceivers will be placed at the 260 m Campus Nord side, meaning the distance between transceivers is  $d = 86.67 \text{ m}$ . The central frequency of operation must be between 2.4 GHz and 2.525 GHz; the middle value 2.4625 GHz is taken.

With all the considered values (12) is solved.

$$P_{RX} [\text{dBm}] = -6 \text{ dBm} + 2 \text{ dB} - 20 \log \left( \frac{4\pi 86.67 \text{ m}}{\frac{3 \times 10^8 \text{ m s}^{-1}}{2.4625 \text{ GHz}}} \right) [\text{dB}] + 2 \text{ dB} - 8 \text{ dB} \quad (13)$$

$$P_{RX} = -87 \text{ dBm}$$

For the nRF24, at 250kbps the sensitivity that achieves  $p_{eb} = 1 \times 10^{-3}$  is  $-94 \text{ dBm} - 10 \text{ dBm} = -104 \text{ dBm}$ . The difference between the received power at the LNA input and the mentioned sensitivity is  $-87 \text{ dBm} - (-104 \text{ dBm}) = 17 \text{ dB}$ .

Now, from Figure 1 and testbed measurements it's derived that to get  $p_{eb} = 1 \times 10^{-3}$ ,  $\frac{E_b}{N_0} = 13 \text{ dB}$ . If there's 17 dB difference between the received power and the sensitivity for which  $p_{eb} = 1 \times 10^{-3}$ , then  $p_{eb}$  is obtained by looking at  $\frac{E_b}{N_0} = 30 \text{ dB}$ .

The testbed measurements plot is quite linear in a logarithmic scale, with an approximate slope of  $\approx 3.5$  decades for every 10 dB increase in  $\frac{E_b}{N_0}$ . Thus, in this numeric case  $p_{eb} \approx 1 \times 10^{-9}$ .

The DATA packet defined in the protocol requires 30 bytes plus 7 bits. This payload is fitted inside the nRF24 packet. This means a preamble of 1 byte, an address of up to 5 bytes, a packet control field of 9 bits and a CRC of up to 2 bytes are also required. This adds up to

$$N = (1 + 5 + 2 + 30) \text{ bytes} \times \frac{8 \text{ bits}}{1 \text{ byte}} + 9 \text{ bits} + 7 \text{ bits} = 320 \text{ bits} \quad (14)$$

Substituting the  $N$  number of bits and the bit error probability  $p_{eb}$  into (3) leads to the following  $p_{ep}$  packet error:

$$p_{ep} = 1 - (1 - p_{eb})^N$$

$$p_{ep} = 1 - \left(1 - 1 \times 10^{-9}\right)^{320} \quad (15)$$

$$p_{ep} = 320 \times 10^{-9}$$

A failure threshold of  $p_{\text{fail threshold}} = 1 \times 10^{-12}$  is imposed and  $T$  is found using (6).

$$\begin{aligned}
 T &> \frac{\ln(p_{\text{fail threshold}})}{\ln(1 - (1 - p_{eb})^N)} \\
 T &> \frac{\ln(1 \times 10^{-12})}{\ln(1 - (1 - 1 \times 10^{-9})^{320})} . \\
 T &> 1.848 \\
 T &\rightarrow 2
 \end{aligned} \tag{16}$$

This means that the number of retries is just

$$R = 1 , \tag{17}$$

due to the low probability of getting a packet wrong. The fact that the 250kbps datarate is set means the sensitivity is very low and leads to a very robust communication for the distances the application requires, according to theoretical calculations.

### 2.3.2. DATA packet, 3 nodes distance

The previous case considered the distance between transmitter and receiver was just  $d = 86.67$  m. Although this is expected to be the most common case, the low sensitivity makes it possible to send data at higher distances. Here,  $d = 260$  m, which is the maximum line of sight (LOS) distance expected, is taken.

From (12), it's enough to calculate the received power decrease because of the distance increase.

$$\begin{aligned}
 \Delta P_{RX} &= -20 \log(d_{\text{current}}) + 20 \log(d_{\text{previous}}) \\
 \Delta P_{RX} &= -20 \log(260 \text{ m}) + 20 \log(86.67 \text{ m}) . \\
 \Delta P_{RX} &= -11.56 \text{ dB}
 \end{aligned} \tag{18}$$

Thus,

$$\begin{aligned}
 P_{RX} &= P_{RX, \text{previous}} + \Delta P_{RX} \\
 P_{RX} &= -87 \text{ dBm} - 11.56 \text{ dB} . \\
 P_{RX} &= -98.564 \text{ dBm}
 \end{aligned} \tag{19}$$

In this case, the difference of the received power with respect to the sensitivity level is  $-98.564 \text{ dBm} - (-104 \text{ dBm}) = 5.44 \text{ dB}$ . Adding up the 13 dB, the Figure 1 must be analyzed at 18.55 dB. In this case, by extrapolating the testbed measurements plot a safe value to take is

$$p_{eb} = 1 \times 10^{-5} . \tag{20}$$

Considering the  $N = 320$  bits of the DATA packet, the packet error probability is:

$$\begin{aligned}
 p_{ep} &= 1 - (1 - p_{eb})^N \\
 p_{ep} &= 1 - (1 - 1 \times 10^{-5})^{320} . \\
 p_{ep} &= 3.195 \times 10^{-3}
 \end{aligned} \tag{21}$$



The difference with the previous case is around 4 orders of magnitude. Again, a failure threshold of  $p_{\text{fail threshold}} = 1 \times 10^{-12}$  is imposed and  $T$  is found using (6).

$$\begin{aligned}
 T &> \frac{\ln(p_{\text{fail threshold}})}{\ln(1 - (1 - p_{eb})^N)} \\
 T &> \frac{\ln(1 \times 10^{-12})}{\ln(1 - (1 - 1 \times 10^{-5})^{320})} . \\
 T &> 4.809 \\
 T &\rightarrow 5
 \end{aligned} \tag{22}$$

This means that the number of retries must be

$$R = 4 . \tag{23}$$

Thanks the exponential dependence of  $p_{ep}$  on the number of tries  $T$  the 4 orders of magnitude between the two cases just require a 3 tries difference.

### 2.3.3. TOKEN packet, neighbour distance

The two previous cases detailed in subsections 2.3.1 and 2.3.2 refer to the DATA packet, which is the largest packet specified by the standard. However, the TOKEN packet should also be considered. As mentioned in the standard, a transmitter needs to make sure if a node is available or not. The TOKEN packet contains a variable part of 3 bits, larger than the 2 bits of the HELLO RESPONSE packet. The TOKEN packet is analyzed because it's the largest of the two.

Thanks to previous calculations the  $P_{RX}$  power are already known. Changes appear, now, at the packet number of bits,  $N$ .

$$N = (1 + 5 + 2) \text{ bytes} \times \frac{8 \text{ bits}}{1 \text{ byte}} + 9 \text{ bits} + 3 \text{ bits} = 67 \text{ bits} . \tag{24}$$

For the neighbour distance

$$P_{RX} = -87 \text{ dBm} . \tag{25}$$

Recall this lead to

$$p_{eb} = 1 \times 10^{-9} . \tag{26}$$

Considering the  $N = 67$  bits of the TOKEN packet, the packet error probability is:

$$\begin{aligned}
 p_{ep} &= 1 - (1 - p_{eb})^N \\
 p_{ep} &= 1 - (1 - 1 \times 10^{-9})^{67} . \\
 p_{ep} &= 67.0 \times 10^{-9}
 \end{aligned} \tag{27}$$

In this case, the packet failure threshold is decreased by two orders of magnitude with respect to the DATA packet, i.e.,  $p_{\text{fail threshold}} = 1 \times 10^{-14}$ . This number is arbitrary, but the authors

want to emphasize the importance of sending the TOKEN packet properly. (6) is solved for  $T$ .

$$\begin{aligned}
 T &> \frac{\ln(p_{\text{fail threshold}})}{\ln(1 - (1 - p_{eb})^N)} \\
 T &> \frac{\ln(1 \times 10^{-14})}{\ln(1 - (1 - 1 \times 10^{-9})^{67})} . \\
 T &> 1.952 \\
 T &\rightarrow 2
 \end{aligned} \tag{28}$$

On one hand, the threshold has been decreased; on the other the number of bits  $N$  is lower. This means that the number of retries must be just 1 for the neighbour distance.

$$R = 1 . \tag{29}$$

#### 2.3.4. TOKEN packet, 3 nodes distance

The main difference with respect to subsection 2.3.3 is that now  $P_{RX} = -98.564$  dBm, which means

$$p_{eb} = 1 \times 10^{-5} . \tag{30}$$

Recall the defined TOKEN packet contains  $N = 67$  bits. Then,

$$\begin{aligned}
 p_{ep} &= 1 - (1 - p_{eb})^N \\
 p_{ep} &= 1 - (1 - 1 \times 10^{-5})^{67} . \\
 p_{ep} &= 669.78 \times 10^{-6}
 \end{aligned} \tag{31}$$

There's been a 4 orders of magnitude increase in  $p_{ep}$  with respect to subsection 2.3.3. Again, the packed failure threshold is set to  $p_{\text{fail threshold}} = 1 \times 10^{-14}$ . (6) is solved for  $T$ .

$$\begin{aligned}
 T &> \frac{\ln(p_{\text{fail threshold}})}{\ln(1 - (1 - p_{eb})^N)} \\
 T &> \frac{\ln(1 \times 10^{-14})}{\ln(1 - (1 - 1 \times 10^{-5})^{67})} . \\
 T &> 4.411 \\
 T &\rightarrow 5
 \end{aligned} \tag{32}$$

On one hand, the threshold has been decreased; on the other the number of bits  $N$  is lower.

$$R = 4 . \tag{33}$$

#### 2.3.5. Summary

The obtained results are summarized below:

| Packet | Distance (m) | $p_{\text{fail}}$ threshold | Number of tries $T$ | Number of retries $R$ |
|--------|--------------|-----------------------------|---------------------|-----------------------|
| DATA   | 86.67 m      | $1 \times 10^{-12}$         | 2                   | 1                     |
| DATA   | 260 m        | $1 \times 10^{-12}$         | 5                   | 4                     |
| TOKEN  | 86.67 m      | $1 \times 10^{-14}$         | 2                   | 1                     |
| TOKEN  | 260 m        | $1 \times 10^{-14}$         | 5                   | 4                     |

Table 1. Number of tries and retries, summary

In conclusion, it's been justified that for both the DATA and TOKEN packets a total of  $T = 5$  tries, i.e.,  $R = 4$  retries are required, to meet the set thresholds for the 260 m distance.

### 3. Timeouts

#### 3.1. Theoretical explanation

As important as the number of tries and retries are, it's also critical to determine the timeouts required for a transmitter to switch to a receiver and wait for acknowledgements (ACKs) from potential receivers that switch to transmitters after they have received a package.

It should be made clear that the timeout is defined here as the maximum loop time in which a transmitter will be stuck waiting for an acknowledgement signal. Too small, and there won't be time to receive the ACK. Too large, and the transmitter will waste time waiting for no signal. This means there's an optimal timeout. It's the aim of this chapter to find it.

Moreover, it can be noticed from the given definition that the timeout is independent of the package length that has been sent. It can, however, be made dependant on the ACK packet length.

The timeout at the receiver side must meet the following equation.

$$\begin{aligned} t_{\text{timeout}} &\geq t_{\text{air}} + t_{\mu\text{C, RX}} + t_{\text{RTAD}} + t_{\text{RX}} + t_{\text{air}} + t_{\mu\text{C, TX}} + t_{\text{RTAD}} \\ t_{\text{timeout}} &\geq 2t_{\text{air}} + t_{\text{RX}} + (t_{\mu\text{C, RX}} + t_{\mu\text{C, TX}}) + 2t_{\text{RTAD}} \end{aligned} \quad (34)$$

Where  $t_{\text{air}}$  is the amount of time the last bit sent by the transmitter TX is received by the receiver TX,  $t_{\text{TX}}$  is the duration of time the receiver requires to send the ACK packet,  $t_{\mu\text{C, RX}}$  and  $t_{\mu\text{C, TX}}$  are the required times at the RF modules microcontrollers to interpret the received packet and take a decision from there, and  $t_{\text{RTAD}}$  is the radio turn around delay, which is the time a transceiver requires to switch from transmitter to receiver or viceversa.

The packets are sent at the speed of light, meaning  $t_{\text{air}}$  is

$$t_{\text{air}} = \frac{d}{c} \quad (35)$$

Where  $d$  is the distance between transmitter and receiver, and  $c$  is the speed of light  $c \approx 3 \times 10^8 \text{ m s}^{-1}$ .

The time required by the receiver to send the ACK packet,  $t_{\text{RX}}$ , is

$$t_{\text{RX}} = \frac{\text{Packet size [bits]}}{\text{Datarate} \left[ \frac{\text{bits}}{\text{s}} \right]} \quad (36)$$

The time spent at the RF module microcontroller,  $t_{\mu\text{C}}$ , is proportional to the clock periods to fetch, decode and execute the required instructions to process the message and take a decision. This depends on the integrated circuit internal operation and clock frequency.

The  $t_{\text{RTAD}}$  time is typically specified by the RF module manufacturer.

### 3.2. Numerical example for the NM competition scenario

Once the timeout equation (34) has been specified, numerical values can be substituted to calculate the timeout required for a real application.

The maximum expected line of sight communication distance during the NM competition is  $d = 260$  m. Thus,

$$\begin{aligned} t_{\text{air}} &= \frac{d}{c} \\ t_{\text{air}} &= \frac{260 \text{ m}}{3 \times 10^8 \text{ m s}^{-1}} \cdot \\ t_{\text{air}} &= 86.67 \text{ ns} \end{aligned} \quad (37)$$

The time required to transmit an ACK package depends on its length. Although all packages defined in the standard have very similar lengths, the largest one is considered here. It's the DATA ACK packet, that shares the length with the HELLO RESPONSE packet. Both have 2 bits of information plus the fixed bytes and bits of the nRF24.

$$N = (1 + 5 + 2) \text{ bytes} \times \frac{8 \text{ bits}}{1 \text{ byte}} + 9 \text{ bits} + 2 \text{ bits} = 66 \text{ bits} \cdot \quad (38)$$

The standard specifies Datarate = 250 kbps.

$$t_{\text{RX}} = \frac{66 [\text{bits}]}{250 \times 10^3 \left[ \frac{\text{bits}}{\text{s}} \right]} = 264 \mu\text{s} \cdot \quad (39)$$

The  $t_{\mu\text{C}}$  times are not specified by Nordic, the nRF24 manufacturer, and one may suspect these times are already included in  $t_{\text{RTAD}}$ . In case of doubt, the following assumption is made:

$$t_{\mu\text{C}, \text{RX}} = t_{\mu\text{C}, \text{TX}} = t_{\mu\text{RTAD}} \cdot \quad (40)$$

According to the nRF24 specifications

$$t_{\text{RTAD}} = 130 \mu\text{s} \cdot \quad (41)$$

Finally, numerical values are plugged into (34).

$$\begin{aligned} t_{\text{timeout}} &\geq 2t_{\text{air}} + t_{\text{RX}} + (t_{\mu\text{C}, \text{RX}} + t_{\mu\text{C}, \text{TX}}) + 2t_{\text{RTAD}} \\ t_{\text{timeout}} &\geq 2 \times 86.67 \text{ ns} + 264 \mu\text{s} + 2 \times 130 \mu\text{s} + 2 \times 130 \mu\text{s} \cdot \\ t_{\text{timeout}} &\geq 784 \mu\text{s} \end{aligned} \quad (42)$$

This is the minimum time required between two transmissions at the transmitter side. There's a high chance the nRF24 already implements some of the timeout sumands. Anyways, the communication should work fine if the above timeout is imposed to wait for all the possible responses.

## Bibliography

- [1] J. R. Luque, M. J. Morón, and E. Casilari, “Analytical and empirical evaluation of the impact of gaussian noise on the modulations employed by bluetooth enhanced data rates,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, no. 1, Mar. 2012. DOI: [10.1186/1687-1499-2012-94](https://doi.org/10.1186/1687-1499-2012-94). [Online]. Available: <https://doi.org/10.1186/1687-1499-2012-94>.